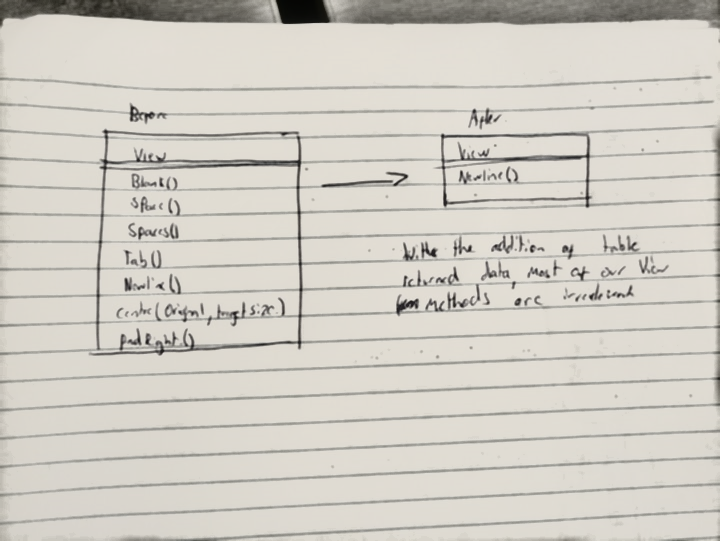
# Iteration 4

## Goal:

Implement Tables to display data neatly

## Class Diagram of View after Leaning and removing Redundancy



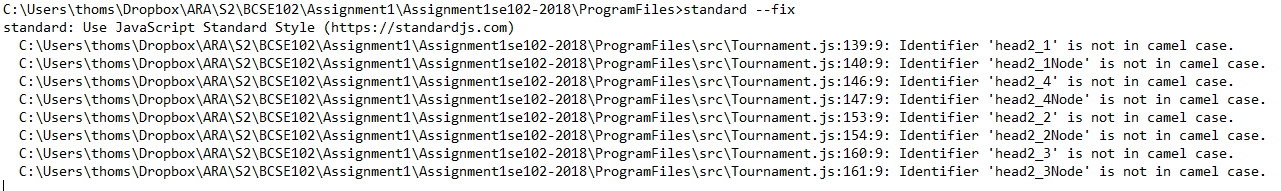
## Tasks:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task | Description | Products | Time est: | Time Taken |
| Planning | * Complete Worksheet and Planning sheet | * Worksheet | 1 hour | 1.5 hours |
| Analysis: | * Create a comparison class diagram for view * Complex algorithm worksheet | * Class Diagram * algorithm | 30mins  30 mins | 30 mins  30mins |
| Designing: | * Create a pseudocode for table functions. | * Pseudocode | 30 mins | 15 mins |
| Building | * Create Callable functions to add table support. * Edit existing sport methods to support tables | * Working assignment (Hopefully!). * Tabularised Data | 30 mins  30 mins | 45 mins  1.5 hours |
| Testing | * Run through standard js | * Tested Web page | 15 mins | 30 mins |

## Standard JS Log

Exceptions listed in the log below are technicalities that the system can’t pick up, not the fact that it isn’t in camel case.

Our new StandardJs log:



## Pseudocode:

## 

## Review:

### Mistakes:

One mistake I made was the addition of extra functions in my tabular data that were not needed. I made assumptions from one area of the functions made and applied it to another without testing, and in turn created redundancy in my code.

I also did not remove redundant functions in my View class. In keeping with SE principal of keeping programs lean, I need to analyses and remove redundant parts of my code that is no longer needed, as I feel like my program files are getting quite complex and unnecessarily large, which will have an impact on page performance.

### What I Will do Different Next Iteration

Test all new functions using a testing table in the future might stop redundancy in my code, as well as testing functionality. This should also help reduce time wasted on coding excess functions.